

A Consensus Protocol under Directed Communications with Two Time Delays and Delay Scheduling

Rudy Cepeda-Gomez* and Nejat Olgac†

Abstract

This paper studies a consensus protocol over a group of agents driven by second order dynamics. The communication among members of the group is assumed to be directed and affected by two rationally independent time delays, one in the position and the other in the velocity information channels. These delays are unknown but considered to be constant and uniform throughout the system. The stability of the consensus protocol is studied using a simplifying factorization procedure and deploying the Cluster Treatment of Characteristic Roots (CTCR) paradigm. This effort results in a unique depiction of the exact stability boundaries in the domain of the delays. CTCR requires the knowledge of the potential stability switching loci exhaustively within this domain. The creation of these loci is an important contribution of this work. It is done in a new surrogate coordinate system, called the *Spectral Delay Space (SDS)*. The relative stability, i.e., the speed to reach consensus of the system is also investigated for this class of systems. Based on the outcome of this effort, a paradoxical control design concept is introduced. It is called the *Delay Scheduling*, which is another key contribution of this paper. It reveals that the performance of the system may be improved by increasing the delays. The amount of increase, however, is only revealed by the CTCR. Example case studies are presented to verify the underlying analytical derivations.

1 Introduction

Recently considerable research effort has been placed in the multi-agent systems area, and particularly in the consensus behaviour of swarms. The consensus aims at an agreement among the agents. This problem is treated by [28] in the early on, where agents aim to align their headings using discrete time representation. Later, [17] introduced a formal framework of the consensus problem for agents driven by first order dynamics. That study considers directed communication topologies, both for fixed and switching types, and introduce some graph theoretical results useful for the stability analysis of such agreement protocols. Under the simplifying features of first order dynamics, they

*Faculty of Mechatronic Engineering, Universidad Santo Tomas, Bucaramanga, Colombia. Email: rudycepedagomez@mail.ustabuca.edu.co

†Mechanical Engineering Department, University of Connecticut, Storrs, CT, USA. Email: olgac@engr.uconn.edu

also consider time delayed communications in the case of a fixed topology. Many other researchers suggest further extensions to this work, proposing consensus protocols for agents driven by second order dynamics [22, 21] without time delays in the communication channels.

The works by Lin and his co-workers include the study of agents driven by second order discrete [11] and continuous [12] dynamics, and applications of different control laws, ranging from a basic P - D -like (proportional and integral) control logic [13], to H_∞ structure [14]. Most of these works entail time delayed communications and directed topologies. Sun and Wang also consider the consensus problem with fixed [26] and time varying delays [25]. [20] study a leader follower case, also with time varying delays. [16] presents a comparison study among several different swarm control laws is, including first and second order agents and considering two time delays.

When it comes to the stability analysis of consensus systems with time delays, almost all the previous works rely on Lyapunov-Krasovskii or Razhumikin based methodologies [11, 12, 13, 14, 16, 17]. They produce only very conservative results, leading to stability bounds for very small delays. Additionally, because these results are based on some solutions of an LMI, they are always imprecise, conservative and, very critically, cumbersome to deploy.

As the non-delayed control structure imparts a stable consensus, conservatively small delays would still maintain the stable posture. Very differently from the general trend, we wish to determine the stability bounds in delay non-conservatively and exhaustively in this paper. This venue opens up an unusual path of control logic design, which is called the *Delay Scheduling*.

Recently the authors have presented a methodology for the analysis of consensus protocols with single and multiple, rationally-independent time delays [5]. This earlier work is based on the combination of a simplifying factorization procedure over the characteristic equation of the system and the deployment of a crucial stability paradigm, which is called the Cluster Treatment of Characteristic Roots, CTCR, [19, 8, 6]. CTCR provides a tool for assessing the stability of LTI systems with multiple rationally-independent delays. This new method uniquely creates exact, exhaustive and explicit stability regions in the domain of the delays. All of our earlier investigations [5] have focused on undirected communications topologies. In the present document, on the other hand, we consider directed topologies. The transition from undirected to directed topologies is not trivial and the treatment of the ensuing difference forms the critical contribution of this document. Although some other investigators have treated this problem, as mentioned before, this is the first work, to the best of our knowledge, in which truly large time delays and exact stability boundaries are handled precisely (non-conservatively) and exhaustively. The "large" delay, here, implies that they are either orders of magnitude larger than the control sampling period or they are in the order of the fastest desired dynamics in the system.

In summary, the two main contributions of this paper are as follows: for a group dynamics to achieve consensus under time-delayed communication topologies:

1. The *delay scheduling* concept is introduced for swarm control for the first time. It

is a direct consequence of the stability assessment paradigm, called CTCR, which declares the exact and exhaustive stable regions in the domain of the two rationally independent time delays.

2. This treatment is deployed for the first time for directed inter-agent communication topologies on a group dynamics with second order governing equations.

Bold face notation is used in the text for vector quantities, bold capital letters for matrices and italic symbols for scalars.

2 Consensus Protocol

We consider a group of n autonomous agents, which are driven by second order dynamics given by $\ddot{x}_j(t) = u_j(t)$, $j = 1, 2, \dots, n$, where $x_j(t) \in \mathbb{R}$ is taken as the scalar position and $u_j(t) \in \mathbb{R}$ as the control law, without loss of generality. The arguments are presented for a one-dimensional case for simplicity, and all the results are scalable to higher dimensions by using the Kronecker product notation, the additional dimensions are decoupled in the governing dynamics. Features such as "holonomicity" and "non-linearity" in the agent dynamics would also prevent this expansion.

Following the convention in the literature [17, 13, 14, 12, 16, 26, 25, 20, 5], we declare the consensus is achieved when all n agents are at the same position, i.e., $\lim_{t \rightarrow \infty} (x_j(t) - x_k(t)) = 0$ for any $j, k \in [1, n]$. Notice that this consensus definition does not state anything *a priori* regarding the value of the final position.

The j -th agent receives position and velocity information from a subset of agents, \mathcal{N}_j , which consists of δ_j agents, $\delta_j < n$, and it is called the *informers* of the j -th agent. Assuming unidirectional channels, the communication network can be described by a directed graph with n vertices, and δ_j is the in-degree of node j . It is also assumed that all these communication channels have two constant and uniform time delays, τ_1 and τ_2 , which affect the position and velocity information exchange, respectively. In order to create the consensus we propose a *PD* (proportional and derivative) decentralized control law for each agent j :

$$\ddot{x}_j(t) = P \left(\sum_{k \in \mathcal{N}_j} \frac{x_k(t - \tau_1)}{\delta_j} - x_j(t) \right) + D \left(\sum_{k \in \mathcal{N}_j} \frac{\dot{x}_k(t - \tau_2)}{\delta_j} - \dot{x}_j(t) \right) \quad (1)$$

where P and D are user-selected, positive control gains. This logic tries to bring the agent's current position to the centroid of its informers, \mathcal{N}_j , and its velocity to the mean velocity of the same set of informers (or the velocity of the same centroid), using the last known positions and velocities of the informer agents. Notice that this protocol does not include self delayed information, while all the data from the informers is delayed, positions by τ_1 and velocities by τ_2 . The corresponding dynamics of the complete system

in state space becomes:

$$\begin{aligned} \dot{\mathbf{x}}(t) = & \left(\mathbf{I}_n \otimes \begin{bmatrix} 0 & 1 \\ -P & -D \end{bmatrix} \right) \mathbf{x}(t) \\ & + \left(\mathbf{C} \otimes \begin{bmatrix} 0 & 0 \\ P & 0 \end{bmatrix} \right) \mathbf{x}(t - \tau_1) + \left(\mathbf{C} \otimes \begin{bmatrix} 0 & 0 \\ 0 & D \end{bmatrix} \right) \mathbf{x}(t - \tau_2) \end{aligned} \quad (2)$$

with the state vector being a concatenation of the n positions and velocities: $\mathbf{x} = [x_1 \ \dot{x}_1 \ x_2 \ \dot{x}_2 \ \cdots \ x_n \ \dot{x}_n]^T \in \mathbb{R}^{2n}$. In (2), \otimes denotes Kronecker multiplication [23, see], \mathbf{I}_n is the identity matrix of dimension n and $\mathbf{C} = \mathbf{\Delta}^{-1} \mathbf{A}_\Gamma$ is the weighted adjacency matrix, created by the multiplication of the inverse of $\mathbf{\Delta} = \text{diag}(\delta_1, \delta_2, \dots, \delta_n) \in \mathbb{R}^{n \times n}$, known as the in-degree matrix, and \mathbf{A}_Γ , the adjacency matrix of the communication topology, following graph theory conventions [10, 3, see].

The dynamics in (2) can be expressed in a more compact form as:

$$\dot{\mathbf{x}}(t) = \mathbf{A}\mathbf{x}(t) + \mathbf{B}_1\mathbf{x}(t - \tau_1) + \mathbf{B}_2\mathbf{x}(t - \tau_2) \quad (3)$$

with self evident \mathbf{A} , \mathbf{B}_1 and \mathbf{B}_2 matrices from (2).

The complexity level of this dynamics increases rapidly as the number of agents gets larger, making the stability analysis numerically tedious and intractable. The corresponding characteristic equation of the system is

$$Q(s, P, D, \tau_1, \tau_2) = \det(s\mathbf{I}_{2n} - \mathbf{A} - \mathbf{B}_1e^{-\tau_1 s} - \mathbf{B}_2e^{-\tau_2 s}) = 0 \quad (4)$$

This equation is a $2n$ order quasi-polynomial with commensuracy degree n (i.e., up to $n\tau_1$ and $n\tau_2$ terms appear) and, more critically, the cross-talk between the time delays (i.e., terms like $\tau_1 + \tau_2$) are typically present. The complexity of (4) increases further if a more tortuous communication topology is used (e.g., one with non-uniform delay structure). Such general stability problem of multiple time delay systems is notoriously known to be NP-hard [19, 27]. Even the strongest of the existing mathematical tools to analyse the stability of time delay systems of this class falls short to handle such cases exhaustively. Two important procedures are combined to remedy the impasse *vis-à-vis* the particular consensus problem with directed topologies. First, a factorization operation, introduced by [5], is performed to break the characteristic equation into factors of reduced order quasi-polynomial and simpler but identical forms. The crucial difference of this paper from our earlier work appears here. The directed topologies result in complex conjugate identifiers of the factors, as opposed to all real in the undirected cases. Second, a method, called the Cluster Treatment of Characteristic Roots (CTCR) [19, 8, 6] and a new domain, Spectral Delay Space (SDS), which we describe in Section 3, are deployed for the first time on directed topologies to perform an exact and exhaustive stability analysis. This combination of handling complex eigenvalues and SDS facilitates an efficient and novel mechanism to resolve the stability problem at hand which constitutes the main contribution of the present work.

2.1 Factorization Procedure

The stem procedure in [5], which transforms the characteristic equation (4) into a product of a set of second order factors, was based on the use of an undirected communication structure. For undirected case, the eigenvalues of the matrix \mathbf{C} are always real. When directed topologies are considered, however, the matrix \mathbf{C} may also have complex conjugate eigenvalues. While this does not affect the main philosophy behind the methodology we present, the practical implementation changes substantially.

Assume that there is a non-singular matrix \mathbf{T} which converts \mathbf{C} into its Jordan canonical form: $\mathbf{\Lambda} = \mathbf{T}^{-1}\mathbf{C}\mathbf{T}$. The matrix $\mathbf{\Lambda}$ is block diagonal of the form:

$$\mathbf{\Lambda} = \begin{bmatrix} \lambda_1 & 0 & \cdots & 0 & 0 & \cdots & 0 \\ 0 & \lambda_2 & \cdots & 0 & 0 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \lambda_\ell & 0 & \cdots & 0 \\ \mathbf{0}_{21} & \mathbf{0}_{21} & \cdots & \mathbf{0}_{21} & \mathbf{J}_{\ell+1} & \cdots & \mathbf{0}_{21} \\ \vdots & \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ \mathbf{0}_{21} & \mathbf{0}_{21} & \cdots & \mathbf{0}_{21} & \mathbf{0}_{21} & \cdots & \mathbf{J}_{\ell+m} \end{bmatrix} \quad (5)$$

where $\mathbf{0}_{21} = [0 \ 0]^T$; λ_j , $j = 1, 2, \dots, n$, are the (size 1) Jordan blocks corresponding to the real eigenvalues of \mathbf{C} and

$$\mathbf{J}_j = \begin{bmatrix} \Re(\lambda_j) & -\Im(\lambda_j) \\ \Im(\lambda_j) & \Re(\lambda_j) \end{bmatrix}, \quad j = \ell + 1, \ell + 2, \dots, \ell + m \quad (6)$$

are the 2×2 Jordan blocks corresponding to the complex conjugate eigenvalue pairs.

Remark I. It is trivial to observe that transformation $\mathbf{x}(t) = (\mathbf{I}_2 \otimes \mathbf{T}^{-1}) \boldsymbol{\xi}$ converts (2) into $\ell + m$ decoupled subsystems, which generate the factors of the characteristic equation, as we display next. The subsystems corresponding to real eigenvalues (size 1 Jordan blocks) create factors of the form:

$$q_j(s, P, D, \tau_1, \tau_2, \lambda_j) = s^2 + Ds + P - \lambda_j (Dse^{-\tau_2 s} + Pe^{-\tau_1 s}) \quad (7a)$$

whereas the size two Jordan block corresponding to complex conjugate eigenvalue pairs generate factors like:

$$\begin{aligned} q_j(s, P, D, \tau_1, \tau_2, \lambda_j) = & s^4 + 2Ds^3 + (D^2 + 2P)s^2 + 2DPs + P^2 \\ & - 2\Re(\lambda_j)(s^2 + Ds + P)(Dse^{-\tau_2 s} + Pe^{-\tau_1 s}) + |\lambda_j|^2 (Dse^{-\tau_2 s} + Pe^{-\tau_1 s})^2 \end{aligned} \quad (7b)$$

The characteristic equation of the complete system is then formed by the product of

these $\ell + m$ individual factors.

$$\begin{aligned}
Q(s, P, D, \tau_1, \tau_2) &= \prod_{j=1}^{\ell+m} q_j(s, P, D, \tau_1, \tau_2, \lambda_j) = \\
\prod_{j=1}^{\ell} [s^2 + Ds + P - \lambda_j (Dse^{-\tau_2 s} + Pe^{-\tau_2 s})] &= \prod_{j=\ell+1}^{\ell+m} [s^4 + 2Ds^3 + (D^2 + 2P)s^2 \\
&+ 2DPs + P^2 - 2\Re(\lambda_j)(s^2 + Ds + P)(Dse^{-\tau_2 s} + Pe^{-\tau_2 s}) \\
&+ |\lambda_j|^2 (Dse^{-\tau_2 s} + Pe^{-\tau_2 s})^2] = 0
\end{aligned} \tag{8}$$

Clearly, this procedure simplifies the problem considerably, by transforming it from a $2n$ order system with time delays of commensuracy degree n and delay cross-talk into a combination of ℓ second order and m fourth order subsystems with maximum commensuracy degree of 2 (e.g., $e^{-2\tau_1 s}$) and single delay cross-talk (e.g., $e^{-(\tau_1+\tau_2)s}$). Furthermore, since the only differentiating element from one factor to the other is the eigenvalue λ_j , the stability analysis in the domain of the delays can be performed just once in each case, for a generic real λ and a generic complex λ . The successive repetitions of the results from this for $\ell + m$ different subsystems will suffice to obtain the stability picture of the complete system. This feature provides us a very strong computational platform for the stability analysis.

Remark II. We now wish to direct the discussions to some other features of the eigenvalues of the weighted adjacency matrix \mathbf{C} . From the way in which this matrix is created, its diagonal consists of zeros only, and any row elements always add up to 1. The latter property makes \mathbf{C} a row-stochastic matrix of which the components are all non-negative [15]. Row-stochastic non-negative matrices possess a wonderful feature: the eigenvalues of such matrices are always within the unit disk of the complex plane. This feature arises as an extension to the Gershgorin's disk theorem [2].

Remark III. Furthermore, it is proven by [1] that if the topology is connected and has at least one spanning tree, $\lambda = 1$ is one of the eigenvalues of the weighted adjacency matrix, \mathbf{C} with multiplicity 1. Then, there is a corresponding factor

$$q_j(s, P, D, \tau_1, \tau_2, \lambda_1 = 1) = s^2 + Ds + P - (Dse^{-\tau_2 s} + Pe^{-\tau_2 s}) \tag{9}$$

in the characteristic quasi-polynomial (4). Without loss of generality, we will assign this eigenvalue to the state ξ_1 . It can also be shown that the normalized eigenvector corresponding to ξ_1 is always $\mathbf{t}_1 = 1/\sqrt{n} [1 \ 1 \ \dots \ 1]^T \in \mathbb{R}^n$, and it is selected as the first column of the earlier defined transformation matrix \mathbf{T} . We call this quantity, ξ_1 , the weighted centroid, which is, obviously, topology dependent. It represents an agreement dynamics in the swarm. That is, if there is consensus we expect the behaviour of the weighted centroid to represent that agreement. The other factors of the characteristic equation in (8) must then be related to some disagreement dynamics of the system. When these disagreement dynamics are stable (i.e., they vanish asymptotically) the agents should reach a consensus.

It is clear that $s = 0$ is always a stationary root of (9) independent of the delays, τ_1 and τ_2 , which implies that the dynamics of the weighted centroid is at best marginally stable. In fact, provided that the disagreement dynamics is stable, ξ_1 reaches a non-zero steady state, when the topology is connected and has at least one spanning tree (a.k.a. arborescence) [9]. That is, each agent is connected to the root of its arborescence.

If the communication topology does not have a spanning tree, however, $\lambda = 1$ is a multiple eigenvalue of \mathbf{C} and equation (9) is a repeated factor within (8). These factors represent the dynamics of the centroids of the subgroups generated by the subgraphs which contain different spanning trees. If all the disagreement factors are stable the swarm members within a subgroup reach some stationary positions which are generally different. Thus the consensus is not achieved. These facts are stated in the following lemma.

Lemma 1 Group behaviour. *Assume the communication topology has at least one spanning tree. Then, the agents in the group reach a consensus if and only if the factor (9) is marginally stable and all the remaining factors of (9) are stable. Furthermore, the consensus value will be $\sqrt{n}\xi_1(t \rightarrow \infty)$, whereas the other states $\xi_j(t \rightarrow \infty) = 0$ for $j = 2, 3, \dots, n$.*

Proof: First, we prove the necessity condition. From the definition of the state ξ we have $[\xi_1(t) \ \xi_2(t) \ \dots \ \xi_n(t)]^T = \mathbf{T} [x_1(t) \ x_2(t) \ \dots \ x_n(t)]$. If consensus is reached, the agents have a common steady state value which we denote by $\bar{x} = \lim_{t \rightarrow \infty} x_j(t)$, $j = 1, 2, 3, \dots, n$. Then:

$$\lim_{t \rightarrow \infty} [\xi_1(t) \ \xi_2(t) \ \dots \ \xi_n(t)]^T = \bar{x} \mathbf{T}^{-1} [1 \ 1 \ \dots \ 1]^T \quad (10)$$

Since the communication topology is assumed to have a spanning tree, $\lambda_1 = 1$ is a simple eigenvalue of \mathbf{C} , corresponding to the eigenvector $\mathbf{t}_1 = 1/\sqrt{n} [1 \ 1 \ \dots \ 1]^T$, the first column of the transformation matrix \mathbf{T} . Since $\mathbf{T}^{-1} [1 \ 1 \ \dots \ 1]^T = \sqrt{n} \mathbf{T}^{-1} \mathbf{t}_1 = \sqrt{n} [1 \ 0 \ \dots \ 0]^T$, equation (10) leads to $\lim_{t \rightarrow \infty} \xi_1(t) = \sqrt{n} \bar{x}$, which indicates marginal stability for (9) and $\lim_{t \rightarrow \infty} \xi_j(t) = 0$ for $j = 2, 3, \dots, n$ indicating asymptotic stability in the other factors of (8).

Next the sufficiency clause is proven. If (9) is marginally stable (due to the characteristic root at the origin) and all the other factors in (8) are stable, the steady state value of ξ_1 will be constant whereas the remaining $\xi_j(t)$, $j = 2, 3, \dots, n$ tend to zero. Then, $\lim_{t \rightarrow \infty} [\xi_1(t) \ \xi_2(t) \ \dots \ \xi_n(t)]^T = [\sqrt{n} \bar{x} \ 0 \ \dots \ 0]^T$. Using the inverse transformation, the original states become:

$$\lim_{t \rightarrow \infty} [x_1(t) \ x_2(t) \ \dots \ x_n(t)]^T = \mathbf{T} [\bar{x} \ 0 \ \dots \ 0]^T = \bar{x} \mathbf{t}_1 = [\bar{x} \ \bar{x} \ \dots \ \bar{x}]^T \quad (11)$$

implying the agents reach consensus. \square

According to Lemma 1, a stable swarm consensus is reached if all the factors $q_j(s, P, D, \tau_1, \tau_2, \lambda_j)$, with $j = 2, 3, \dots, \ell + m$, and the weighted centroid behaviour $q_1(s, P, D, \tau_1, \tau_2, \lambda = 1)$ are all stable. In the contrary, if all the factors of (9) are stable except q_1 in (8), i.e., the weighted centroid of the group has an unstable behaviour, the agents will move in coherence but along an unstable trajectory.

3 CTCR Paradigm and the novel SDS domain

We just showed that the stability of the swarm dynamics is only achieved provided that each of the factors given in (7a) and (7b) represent stable system behavior. For a given set of control parameters (P, D) these factors have the general formations of:

$$q(s, \tau_1, \tau_2) = g_{11}(s) + g_{12}(s)e^{-\tau_1 s} + g_{13}(s)e^{-\tau_2 s} \quad (12)$$

for (7a), and

$$q(s, \tau_1, \tau_2) = g_{21}(s) + g_{22}(s)e^{-\tau_1 s} + g_{23}(s)e^{-\tau_2 s} + g_{24}(s)e^{-(\tau_1+\tau_2)s} + g_{25}(s)e^{-2\tau_2 s} + g_{26}(s)e^{-2\tau_1 s} \quad (13)$$

for (7b). In fact, by definitions of (7a) and (7b), the class of quasi-polynomials given in (12) is a subset of those in (13). Therefore we will concentrate on the stability treatment of the latter generic class. Notice that this quasi-polynomial represents a system with multiple and rationally independent time delays with delay cross-talk and commensurate degree 2.

The stability analysis of the factors in (13) is not trivial, as the infinite dimensionality is doubled due to the presence of two rationally independent delays. This task is performed deploying a unique methodology called the Cluster Treatment of Characteristic Roots (CTCR) [19, 8, 6]. The main philosophy behind it is the detection of the right half plane characteristic roots (i.e., unstable roots), of a Linear Time Invariant - Multiple Time Delay System (LTI-MTDS) (12). It is well known in the linear systems literature that the number of unstable roots can change only along certain loci (in this case in the domain of the delays). The CTCR method requires the exhaustive knowledge of these, so called, *stability switching curves*. In this paper we utilize a novel approach to obtain this knowledge in a new domain which is called the Spectral Delay Space (SDS). The following paragraphs present some preparatory definitions and explain the key propositions of CTCR, in light of the works of [19] and [8].

Definition 1 Kernel hypercurves \wp_0 : *The hypercurves in the \mathbb{R}^{2+} domain that consist exhaustively of all the points $(\tau_1, \tau_2) \in \mathbb{R}^{2+}$ which cause an imaginary root of (13) at $s = \pm\omega i$ and satisfy the constraint $0 < \tau_k \omega < 2\pi$, $k = 1, 2$, are called the kernel hypercurves. The points on these hypercurves contain the smallest possible delay values that create the given pair of imaginary roots at the frequency ω .*

Definition 2 Offspring hypercurves \wp : *The hypercurves obtained from the kernel by the following pointwise non-linear transformation:*

$$\left\langle \tau_1 + \frac{2\pi}{\omega}j_1, \tau_2 + \frac{2\pi}{\omega}j_2 \right\rangle, \quad j_1, j_2 = 1, 2, \dots \quad (14)$$

are called the offspring hypercurves. They are created by the periodicity of the imaginary roots with respect to the time delay.

Jointly the kernel and the offspring hypercurves create all the loci in the delay space $(\tau_1, \tau_2) \in \mathbb{R}^2$, where the system in (4) has imaginary characteristic roots. this complete set constitutes the departure point of the CTCR paradigm. It is crucial that an appropriate computational tool is used to capture all these hypercurves exhaustively. One should, however, not confuse this procedure with the CTCR paradigm itself, which uniquely declares the stability region exhaustively. for instance in a recent article, [24] fall in this fallacy. They misrepresent a novel procedure of determining the kernel and offspring hyperplanes (they call the *potential stability switching curves* instead) as a "new" stability declaration method, despite the fact that the CTCR is still the umbrella paradigm. This fallacy, indeed, results in the completely incorrect declarations of stability regions for the graphically displayed case study in [24], as well as misleading the readers on the novelties of the work.

We now present one more definition before the two key propositions of CTCR.

Definition 3 Root Tendency, *RT*: At any point $\tau \in \wp_0 \cup \wp$ an infinitesimal increase in any of the individual delays, τ_j , creates a transition of the root. Such transition can be to the right or to the left half of the complex plane. The Root Tendency, *RT*, indicates the direction of this transition as only one of the delays, τ_j , increases by ϵ , $0 < \epsilon \ll 1$, while all the others remain constant:

$$RT|_{s=\omega i}^{\tau_j} = \text{sgn} \left[\Re \left(\frac{\partial s}{\partial \tau_j} \Big|_{s=\omega i} \right) \right] \quad (15)$$

Clearly these root tendencies are -1 for stabilizing and $+1$ for destabilizing root crossings on the imaginary axis.

Proposition 1 Finite Number of Kernel Hypercurves *A given LTI-MTDS can exhibit only a finite number, m , of kernel hypersurfaces. This number is upperbounded by the square of the order of the system: $m < n^2$.*

Proposition 2 Invariance of the Root Tendency *Take an imaginary characteristic root, $s = \omega i$, caused by any one of the infinitely many grid points on the kernel and offspring hypercurves in $(\tau_1, \tau_2) \in \mathbb{R}^{2+}$. The root tendency of these imaginary roots remains invariant so long as the grid points on different offspring hypersurfaces are obtained by keeping one of the delays fixed. That is, the root tendency with respect to the variations of τ_j is invariant from the kernel to the corresponding offspring, obtained by the non-linear mapping (14), as the other delay is fixed.*

Spectral Delay Space (SDS): We describe a new procedure in this segment for determining the kernel (and offspring) hypercurves. It is developed on a new domain called the Spectral Delay Space, SDS [8]. SDS is defined by the coordinates $\nu_j = \tau_j \omega$ for every point $(\tau_1, \tau_2) \in \mathbb{R}^{2+}$ on the kernel or the offspring hypercurves. This is a conditional mapping: if a delay set (τ_1, τ_2) creates an imaginary root ωi , (i.e., if the point is on the kernel or the offspring hypercurves) then $(\tau_1 \omega, \tau_2 \omega)$ forms a point in the SDS. On the

contrary, (τ_1, τ_2) points that do not generate an imaginary root have no representation in the SDS domain.

The main advantage of SDS is that the representation of the kernel hypercurve in the SDS, denoted as \wp_0^{SDS} , which is named the *building hypercurve*, is confined into a square of edge length 2π (see Definition 1). Then, it is only necessary to explore a finite domain in SDS in order to find the representation of the building hypercurves in SDS. This finite domain is known as the building block (BB), i.e., a square of $2\pi \times 2\pi$ for the case of two rationally independent delays. Similarly we name the corresponding representation of offspring hypercurves in the SDS as the *reflection curves*. An important advantage of SDS domain is that the transitions from the building hypercurves to the reflection hypercurves is achieved simply by translating the building hypercurve by $\pi 2$, as opposed to using the pointwise non-linear transformation (14). We prevent the undesirable shape distortion from kernel to offspring which occurs in the delay space. In the contrary, the reflection curves in the SDS, \wp^{sds} , are generated by simply stacking the BB squares (with \wp_0^{SDS} in it) one on top of each other. Naturally, this property is named the *stackability feature* of the SDS. There are several other intriguing properties of the SDS and BB concepts which can be found in the works of [8] and [7].

With these definitions and propositions, we perform the preparatory step of CTCR. It is the exhaustive determination of all the stability switching curves, for the generic factors $q_j(s, P, D, \tau_1, \tau_2, \lambda_j)$ as in (7) within the semi-infinite quadrant of $(\tau_1, \tau_2) \in \mathbb{R}^{2+}$.

There are different ways in which the characteristic equation can be transformed to the SDS domain to obtain the building hypercurves. The following mathematical procedure, which uses a half-angle tangent substitution, is taken from the appendix of the thesis by [7].

We start by replacing $s = \omega i$ into (13), and then the exponential terms are replaced by:

$$e^{\tau_k \omega i} = \cos(\nu_k) + i \sin(\nu_k), \quad \nu_k = \tau_k \omega, \quad k = 1, 2 \quad (16)$$

Then the sine and cosine functions are expressed in terms of a half-angle tangent substitution:

$$\cos(\nu_k) = \frac{1 - z_k^2}{1 + z_k^2}, \quad \sin(\nu_k) = \frac{2z_k}{1 + z_k^2}, \quad z_k = \tan(\nu_k) \quad (17)$$

Now (13) can be written as a polynomial in ω with complex coefficients c_k parametrized in z_1 and z_2 :

$$q_j(\omega, z_1, z_2) = \sum_{k=0}^2 c_k(P, D, \lambda_j, z_1, z_2) (\omega i)^k = 0 \quad (18)$$

If there is a solution $\omega \in \mathbb{R}^+$ to (18), both its real and imaginary parts must be zero

simultaneously:

$$\Re(q_j(\omega, z_1, z_2)) = \sum_{k=0}^2 f_k(z_1, z_2) \omega^k = 0 \quad (19a)$$

$$\Im(q_j(\omega, z_1, z_2)) = \sum_{k=0}^2 g_k(z_1, z_2) \omega^k = 0 \quad (19b)$$

The condition for (19) to have a common root is simply stated using a Sylvester's matrix:

$$\mathbf{M} = \begin{bmatrix} f_2(z_1, z_2) & f_1(z_1, z_2) & f_0(z_1, z_2) & 0 \\ 0 & f_2(z_1, z_2) & f_1(z_1, z_2) & f_0(z_1, z_2) \\ g_2(z_1, z_2) & g_1(z_1, z_2) & g_0(z_1, z_2) & 0 \\ 0 & g_2(z_1, z_2) & g_1(z_1, z_2) & g_0(z_1, z_2) \end{bmatrix} \quad (20)$$

In order to simultaneously satisfy bot (19a) and (19b), \mathbf{M} should be singular. This results in the following expression in terms of z_1 and z_2 , or ν_1 and ν_2 :

$$\det(\mathbf{M}) = F(z_1, z_2) = F(\tan(\nu_1), \tan(\nu_2)) \quad (21)$$

which constitutes a closed form description of the kernel curves in the SDS (ν_1, ν_2) , i.e., the Building Block curves. To obtain its graphical depiction, one of the parameters, say ν_2 , can be scanned in the range of $[0, 2\pi]$ and the corresponding ν_1 values are calculated again in the same range. Notice that every point (ν_1, ν_2) on these curves brings an imaginary characteristic root at $\pm\omega i$. That is, we have a continuous sequence of (ν_1, ν_2, ω) sets all along the building block curves.

In order to assess the stability properties of the system in the space of the delay, we now back transform the building and reflection hypercurves from the (ν_1, ν_2) domain of the SDS to the (τ_1, τ_2) delay space, using the inverse transformation of (16) and generate the kernel and offspring hypercurves. The application of the root tendency invariance property follows, resulting in the complete and exact stability outlook of the system, which is unique. Examples of this construction are presented in the following section.

4 Illustrative Examples

In this section we present the simulation results that verify the theoretical analysis of previous sections. All of the examples use the communication topology of Fig. 1. The corresponding weighted adjacency matrix is:

$$\mathbf{C} = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 \\ 0.5 & 0 & 0 & 0.5 & 0 \\ 0 & 0.5 & 0 & 0.5 & 0 \\ 0 & 0 & 0.5 & 0 & 0.5 \\ 0 & 0.5 & 0.5 & 0 & 0 \end{bmatrix} \quad (22)$$

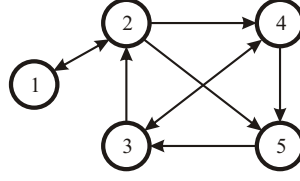


Figure 1: Example directed communication topology with 5 agents.

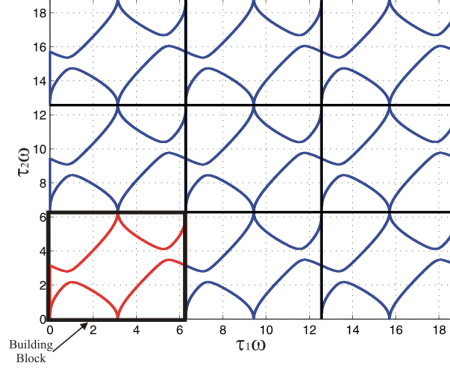


Figure 2: Delay Space representation of the stability switching curves generated by the factor corresponding to $\lambda = 1$ with $P = 2$ and $D = 0.8$.

and its eigenvalues are 1, 0.38, -0.5 , and $-0.44 \pm 0.37i$. The control gains used in the examples are $P = 2$ and $D = 0.8$.

The deployment of the CTCR paradigm using the building block concept leads to Figs. 2 and 3. Figure 2 shows the spectral delay space (SDS) representation of the characteristic factor corresponding to $\lambda = 1$, with the respective building and reflection hypercurves. Other factors are excluded to prevent congestion. The properties of the SDS are clear: the reflection curves, presented in blue, are obtained by translating the building curves, which are depicted in red. Notice that the building curves are confined to a square of side length 2π i.e., the building block.

The superposition of the stability regions of the individual factors results in the stability chart of Fig. 3. Delay combinations inside the shaded portion of this figure result in stable behavior, i.e., agents reach consensus. These results are verified in Fig. 4, in which panel (a) shows the simulation results for a delay combination of $(\tau_1, \tau_2) = (0.5, 0.5)$ seconds, corresponding to point **a** in Fig. 3, panel (b) is for $(\tau_1, \tau_2) = (1, 2.5)$ (point **b** in Fig. 3) and panel (c) is for $(\tau_1, \tau_2) = (1.3, 4.5)$, represented by point **c** in Fig. 3. It is clear that the delay combinations corresponding to the points inside the shaded region (**a** and **c**) generate stable consensus, whereas points outside create divergent behavior.

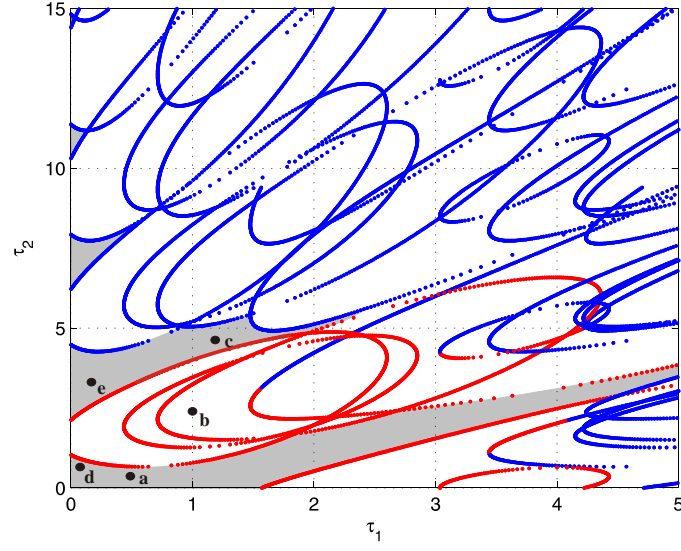


Figure 3: Complete stability picture for the communication topology of Fig. 1, using $P = 2$ and $D = 0.8$.

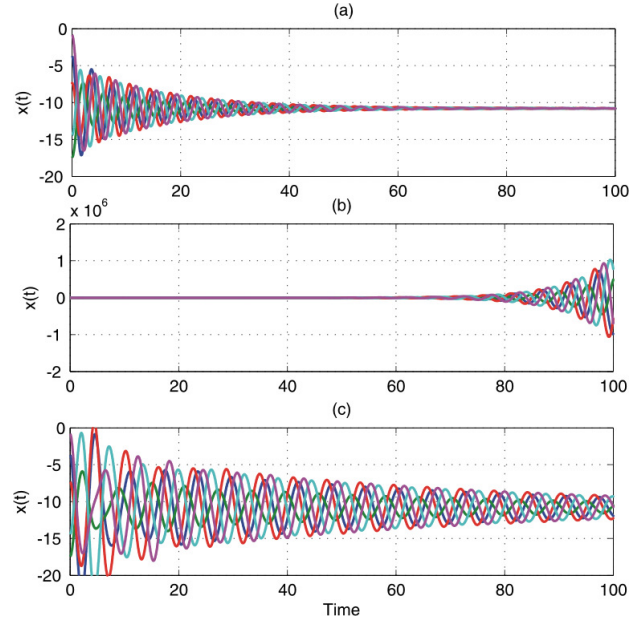


Figure 4: Example agents' behavior for three different points in Fig. 3. From top to bottom, points **a**, **b** and **c**.

4.1 Delay Scheduling Concept

A more interesting, and counter-intuitive, conclusion we can draw from Figs. 3 and 4 is that larger time delays not always imply worsening control performance of the system. Both time delays are larger in **c** than in **b**, however, the system is stable in the first case and unstable in the second. This feature has been used in a control synthesis scheme named the *delay scheduling* [18]. It offers wonderful options for the most effective control parameter selections, because we can manipulate the feedback delay as a part of the control design. Due to the causality, the delay can only be increased (i.e., one cannot sense the events before they become available) and the control designer can, however, intentionally prolong a given set of delays further. This is a paradoxical observation. Below we present a brief discussion on how one should perform the selection of increase in delay, a.k.a. the delay scheduling, along with a deployment scheme of this concept in group dynamics.

We first establish the stability chart of the particular topology following the steps described in the earlier sections (i.e., Fig. 3). Then the present inherently delay composition is monitored in real time. Let us assume these delays render the operating point of **b** in Fig. 3 which is in the unstable region. The controller can now impose an intentional extra delay, as a remedy, to bring the combined delay values to point **c**, where the stability is recovered. It is obvious that such a point must be selected within the stable region. How to select the point **c** in the stable operating region is, however, still an open question. In order to answer this question one needs a crisp stability picture, such as Fig. 3, for which CTCR is the only procedure available today.

The challenge of determining where to take the operating point by introducing extra delays (i.e., the choice of point **c**) offers an additional freedom to the control designer. We aim, here, to select a stable point which facilitates the fastest disturbance rejection capabilities to the swarm formation. As such we pose the operational question of the speed for reaching consensus. This information is crucial in tuning the delays (i.e., prolonging) in order to achieve the most desirable (expeditious) speed in reaching consensus. We explain the relevant steps next on how the stability map, which is given in Fig. 3, can be utilized for this purpose.

The speed of group consensus is dictated by the dominant characteristic root of the infinite dimensional system given in (2). The dominant root of the characteristic equation (4) is the rightmost of the dominant roots of all the factors given. These factors, $q_j(s, P, D, \tau_1, \tau_2, \lambda_j)$, are quasi-polynomials and their roots can only be determined by some numerical approximations, like those proposed by [29] and [4]. We deploy the QPmR (Quasi-Polynomial, mapping-based Root finding) algorithm of [29] here. For each point (τ_1, τ_2) we determine the dominant root and its real part, $\Re(s_{dom})$. For the specific structure in this example case study, it is displayed in Fig. 5.

We wish to stress several observations over this figure:

1. The stable operating zones of Fig. 3 are approximately generated by the point-wise evaluation of the dominant roots over a grid in (τ_1, τ_2) space. Obviously they are the regions where $\Re(s_{dom}) < 0$.

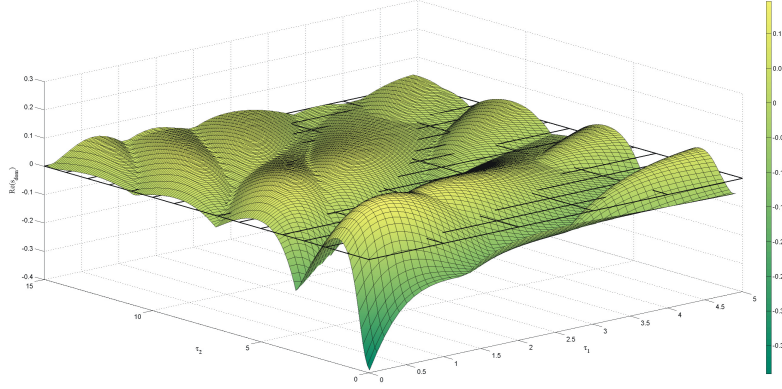


Figure 5: Variation of the real part of the dominant root for the complete system. Grid represents the zero real part partitioning.

2. The time constants expected in reaching consensus, $-1/\Re(s_{dom})$, in example case **a** of Fig. 3 concurs with the location of s_{dom} . For this point $\Re(s_{dom}) = -0.0610$, which corresponds to a time constant of 16.3934 seconds, and a settling time (the speed of reaching consensus) of approximately 66s, as observed in Fig. 4.
3. Although the fastest consensus is achieved at $\tau_1 = \tau_2 = 0$, larger delays do not always mean longer consensus time. Compare for instance points **d** and **e** in Fig. 3. For the first one, $(\tau_1, \tau_2) = (0.05, 0.8)$ and $\Re(s_{dom}) = -0.04$, whereas for **e** $(\tau_1, \tau_2) = (0.1, 3.5)$ and $\Re(s_{dom}) = -0.05$. Obviously, the latter makes a better choice from the point of speed in reaching consensus, although both delays are larger in the second case. This is the core concept of delay scheduling [26-28].

In order to trigger the delay selections under this methodology, one has to have the complete system stability tableau in the delay space, that is, Figs.3 and 5. The procedures which are discussed in this paper to obtain the stability outlook, offer a very efficient technique. For instance, to create one of the individual stability tableaus for a given factor in (8) we encounter an average computational cost of 1s of CPU time, including display operations, on a 2.93 GHz Intel Core 2 Duo - based computer with 4 GB RAM, running Matlab 2010a without an elaborate code optimization for speed.

5 Conclusions

In this paper we study the consensus problem for multi-agent systems, considering directed communication topologies and multiple time delays. Two constant and rationally independent time delays are taken into account, one affecting the position and the other the velocity information exchange among the agents. Both delays can be large and they are assumed to be rationally independent.

The procedure starts with an enabling factorization of the characteristic equation of the system. The factors involved are only distinguished from one another by a set of

eigenvalues of a particular system matrix. The process simplifies the stability analysis considerably and converts it into a repetitive application of a reduced-order core stability analysis n times. The Cluster Treatment of Characteristic Roots (CTCR) methodology and the new concept of Spectral Delay Space are utilized to obtain the exact stability bounds in the domain of the time delays. As the number of agents increase, the only procedural complexity is in determining the set of eigenvalues of a given system matrix. Numerical example cases show the validity of the results.

We also look at the variations on the speed of reaching consensus as the two delays in the communication channels vary. It becomes clear that one may have the option to intentionally prolong the delays to achieve a faster consensus (a procedure called the Delay Scheduling). This ability becomes available only if precise knowledge of the stability picture in the parametric space is available. That is what the CTCR paradigm uniquely provides.

References

- [1] R. Agaev and P. Chebotarev. On the spectra of nonsymmetric laplacian matrices. *Linear Algebra and its Applications*, 399:157–168, 2005.
- [2] H. E. Bell. Gershgorin’s theorem and the zeros of polynomials. *American Mathematical Monthly*, 74:292–295, 1965.
- [3] N. Biggs. *Algebraic Graph Theory*. Cambridge University Press, New York, 1993.
- [4] D. Breda, S. Masset, and A.R. Vermiglio. Pseudospectral differencing methods for characteristic roots of delay differential equations. *SIAM Journal on Scientific Computing*, 27(2):482–495, 2006.
- [5] R. Cepeda-Gomez and N. Olgac. An exact method for the stability analysis of linear consensus protocols with time delay. *IEEE Transactions on Automatic Control*, 56(7):1734–1740, July 2011.
- [6] A. F. Ergenc, N. Olgac, and H. Fazelinia. Extended kronecker summation for cluster treatment of lti systems with multiple delays. *SIAM Journal on Control and Optimization*, 46(1):143–155, 2007.
- [7] H. Fazelinia. *A Novel Stability Analysis of Systems with Multiple Time Delays and Its Application to High Speed Milling Chatter*. Phd. dissertation, University of Connecticut, 2007.
- [8] H. Fazelinia, R. Sipahi, and N. Olgac. Stability robustness analysis of multiple time-delayed systems using building block concept. *IEEE Transactions on Automatic Control*, 52(5):799–810, 2007.
- [9] H. N. Gabow and E. W. Myers. Finding all spanning trees of directed and undirected graphs. *SIAM Journal on Computing*, 7:280–287, 1978.

- [10] C. Godsil and G. Royle. *Algebraic Graph Theory*. Springer, New York, 2004.
- [11] P. Lin and Y. Jia. Consensus of second-order discrete multi-agent systems with nonuniform time delays and dynamically changing topologies. *Automatica*, 45(8):2154–2158, 2009.
- [12] P. Lin and Y. Jia. Further results on decentralised coordination in networks of agents with second-order dynamics. *IET Control Theory and Applications*, 3(7):957–970, 2009.
- [13] P. Lin, Y. Jia, J. Du, and S. Yuan. Distributed control of multi-agent systems with second-order agent dynamics and delay-dependent communications. *Asian Journal of Control*, 10(2):254–259, 2008.
- [14] P. Lin, Y. Jia, and L. Li. Distributed robust h_∞ consensus control in directed networks of agents with time-delay. *Systems and Control Letters*, 57(8):643–653, 2008.
- [15] M. Marcus and H. Minc. *A Survey of Matrix Theory and Matrix Inequalities*. Dover, 1996.
- [16] Z. Meng, W. Ren, Y. Cao, and Y. Zheng. Leaderless and leader-follower consensus with communications and input delays under a directed network topology. *IEEE Transactions on Systems, Man and Cybernetics - Part B*, 41(1):75–88, 2011.
- [17] R. Olfati-Saber and R.M. Murray. Consensus problems in networks of agents with switching topology and time-delays. *IEEE Transactions on Automatic Control*, 49(9):1520–1533, 2004.
- [18] N. Olgac and M.E. Cavdaroglu. Full state feedback controller design with ”‘delay scheduling’” for cart-and pendulum dynamics. *Mechatronics*, 21(1):38–47, 2011.
- [19] N. Olgac and R. Sipahi. Complete stability robustness of third order lti multiple time delay systems. *Automatica*, 41(8):1413–1422, 2005.
- [20] K. Peng and Y. Yang. Leader-following consensus problems with a varying velocity leader and time-varying delays. *Physica A: Statistical Mechanics and its applications*, 388(2-3):193–208, 2009.
- [21] W. Ren. Consensus strategies for cooperative control of vehicle formations. *IET Control Theory and Applications*, 1(2):505–512, 2007.
- [22] W. Ren and R. Beard. Consensus seeking in multi-agent systems under dynamically changing interaction topologies. *IEEE Transactions on Automatic Control*, 50(5):655–661, 2005.
- [23] R. D. Schaefer. *An Introduction to Nonassociative Algebras*. Dover, 1996.

- [24] R. Sipahi and I. I. Delice. Advanced clustering using frequency sweeping methodology for the stability analysis of multiple time delay systems. *IEEE Transactions on Automatic Control*, 56:467–472, 2011.
- [25] Y. Sun and L. Wang. Consensus of multi-agent systems in directed networks with nonuniform time-varying delay. *IEEE Transactions on Automatic Control*, 54(7):1607–1613, 2009.
- [26] Y. Sun and L. Wang. Consensus problems in networks of agents with double integrator dynamics and time delays. *International Journal of Control*, 82(10):1937–1945, 2009.
- [27] R. Toker and H. Ozbay. Complexity issues in robust stability of linear delay-differential systems. *Math. of Control, Signals and Systems*, 9(4):386–400, 1196.
- [28] T. Vicsek, A. Czirók, E. Ben-Jacob, I Cohen, and O. Shochet. Novel type of phase transition in a system of self-driven particles. *Physical Review Letters*, 75(6):1226–1229, 1995.
- [29] T. Vyhlídal and P. Zítek. Mapping based algorithm for large scale computation of quasi polynomial zeros. *IEEE Transactions on Automatic Control*, 54(1):171–177, 2009.